CS 105: DIC on Discrete Structures

Graph theory

Basic terminology, Bipartite graphs and a characterization

Lecture 27

Oct 19 2023

Some simple types of Graphs

- We have already seen some: connected graphs.

Some simple types of Graphs

- We have already seen some: connected graphs.
- paths, cycles.

Some simple types of Graphs

- We have already seen some: connected graphs.
- paths, cycles.
- Are there other interesting classes of graphs?

Bipartite graphs

Definition

A graph is called bipartite, if the vertices of the graph can be partitioned into $V = X \cup Y$, $X \cap Y = \emptyset$ s.t., $\forall e = (u, v) \in E$,

- either $u \in X$ and $v \in Y$
- or $v \in X$ and $u \in Y$

Example: m jobs and n people, k courses and ℓ students.

- How can we check if a graph is bipartite?
- Can we characterize bipartite graphs?

Characterizing bipartite graphs using cycles.

- Recall: A path or a cycle has length n if the number of edges in it is n.
- A path (or cycle) is call odd (or even) if its length is odd (or even, respectively).

Exercise: Prove or Disprove:

Every closed odd walk contains an odd cycle.

Characterizing bipartite graphs using cycles.

Exercise: Prove or Disprove:

Every closed odd walk contains an odd cycle.

Proof: By induction on the length of the given closed odd walk.
Exercise!

Characterizing bipartite graphs using cycles.

Lemma

Every closed odd walk contains an odd cycle.

Theorem, Konig, 1936

A graph is bipartite iff it has no odd cycle.

Proof:

- ($\Longrightarrow$) direction is easy.

Characterizing bipartite graphs using cycles.

Lemma

Every closed odd walk contains an odd cycle.

Theorem, Konig, 1936

A graph is bipartite iff it has no odd cycle.

Proof:

- ($\Longrightarrow$) direction is easy.
- Let G be bipartite with $(V = X \cup Y)$. Then, every walk in G alternates between X, Y.

Characterizing bipartite graphs using cycles.

Lemma

Every closed odd walk contains an odd cycle.

Theorem, Konig, 1936

A graph is bipartite iff it has no odd cycle.

Proof:

- ($\implies$) direction is easy.
- Let G be bipartite with $(V = X \cup Y)$. Then, every walk in G alternates between X, Y.

$\implies$ if we start from X, each return to X can only happen after an even number of steps.

$\implies$ G has no odd cycles.

Characterizing bipartite graphs using cycles.

Lemma

Every closed odd walk contains an odd cycle.

Theorem, Konig, 1936

A graph is bipartite iff it has no odd cycle.

Proof:

- ($\Longleftarrow$) Suppose G has no odd cycle, then let us construct the bipartition. Wlog assume G is connected.

Characterizing bipartite graphs using cycles.

Lemma

Every closed odd walk contains an odd cycle.

Theorem, Konig, 1936

A graph is bipartite iff it has no odd cycle.

Proof:

- ($\Longleftarrow$) Suppose G has no odd cycle, then let us construct the bipartition. Wlog assume G is connected.
- Let $u \in V$. Break V into

$$X = \{v \in V \mid \text{length of shortest path } P_{uv} \text{ from } u \text{ to } v \text{ is even}\},$$
$$Y = \{v \in V \mid \text{length of shortest path } P_{uv} \text{ from } u \text{ to } v \text{ is odd}\},$$

Characterizing bipartite graphs using cycles.

Lemma

Every closed odd walk contains an odd cycle.

Theorem, Konig, 1936

A graph is bipartite iff it has no odd cycle.

Proof:

- ($\Longleftarrow$) Suppose G has no odd cycle, then let us construct the bipartition. Wlog assume G is connected.
- Let $u \in V$. Break V into

$$\begin{aligned} X &= \{v \in V \mid \text{length of shortest path } P_{uv} \text{ from } u \text{ to } v \text{ is even}\}, \\ Y &= \{v \in V \mid \text{length of shortest path } P_{uv} \text{ from } u \text{ to } v \text{ is odd}\}, \end{aligned}$$

- If there is an edge vv' between two vertices of X or two vertices of Y, this creates a closed odd walk: $uP_{uv}vv'P_{v'u}u$.

Characterizing bipartite graphs using cycles.

Lemma

Every closed odd walk contains an odd cycle.

Theorem, Konig, 1936

A graph is bipartite iff it has no odd cycle.

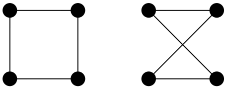
Proof:

- ($\Longleftarrow$) Suppose G has no odd cycle, then let us construct the bipartition. Wlog assume G is connected.
- Let $u \in V$. Break V into

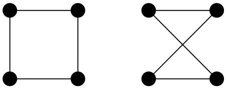
$$\begin{aligned} X &= \{v \in V \mid \text{length of shortest path } P_{uv} \text{ from } u \text{ to } v \text{ is even}\}, \\ Y &= \{v \in V \mid \text{length of shortest path } P_{uv} \text{ from } u \text{ to } v \text{ is odd}\}, \end{aligned}$$

- If there is an edge vv' between two vertices of X or two vertices of Y, this creates a closed odd walk: $uP_{uv}vv'P_{v'u}u$.
- By Lemma, it must contain an odd cycle: contradiction.
- This along with $X \cap Y = \emptyset$ and $X \cup Y = V$, implies X, Y is a bipartition. $\square$

Are these graphs the same?

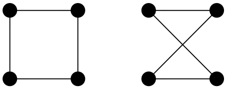


Are these graphs the same?

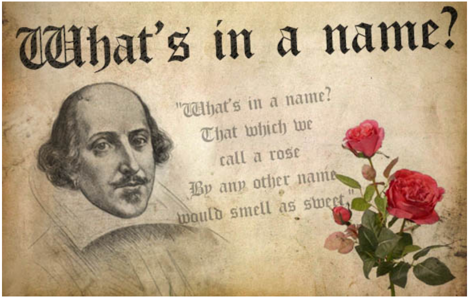


- To compare graphs, we need to name them!

Are these graphs the same?

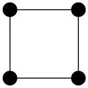


- To compare graphs, we need to name them!



Representing and comparing graphs

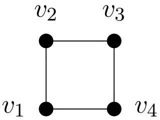
We start with simple graphs...



To represent it, we need to name the vertices...

Representing and comparing graphs

We start with simple graphs...



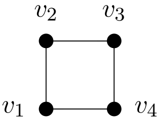
To represent it, we need to name the vertices...

- As an adjacency list:

v_1	v_2, v_4
v_2	v_1, v_3
v_3	v_2, v_4
v_4	v_1, v_3

Representing and comparing graphs

We start with simple graphs...



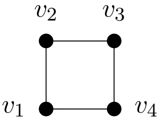
To represent it, we need to name the vertices...

- As an adjacency matrix:

$$\begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \\ v_4 \end{array} \begin{array}{c} \begin{array}{cccc} v_1 & v_2 & v_3 & v_4 \end{array} \\ \begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix} \end{array}$$

Representing and comparing graphs

We start with simple graphs...



To represent it, we need to name the vertices...

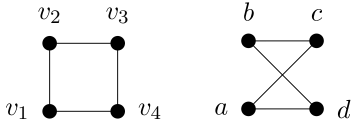
- As an adjacency matrix:

$$\begin{array}{c c} & \begin{array}{cccc} v_1 & v_2 & v_3 & v_4 \end{array} \\ \begin{array}{c} v_1 \\ v_2 \\ v_3 \\ v_4 \end{array} & \begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix} \end{array}$$

- But we want to study properties that are independent of the naming, e.g., connectivity.
- Are two given graphs the “same”, wrt these properties?

Representing and comparing graphs

We start with simple graphs...



To represent it, we need to name the vertices...

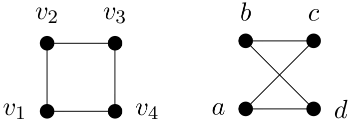
- As an adjacency matrix:

$$\begin{array}{c c} & \begin{array}{cccc} v_1 & v_2 & v_3 & v_4 \end{array} \\ \begin{array}{c} v_1 \\ v_2 \\ v_3 \\ v_4 \end{array} & \begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix} \end{array}$$

- But we want to study properties that are independent of the naming, e.g., connectivity.
- Are two given graphs the "same", wrt these properties?

Representing and comparing graphs

We start with simple graphs...



To represent it, we need to name the vertices...

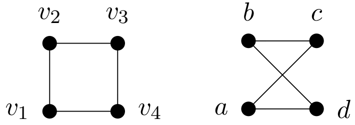
- As an adjacency matrix:

$$\begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \\ v_4 \end{array} \begin{array}{c} \begin{array}{cccc} v_1 & v_2 & v_3 & v_4 \end{array} \\ \begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix} \end{array} \qquad \begin{array}{c} \\ a \\ b \\ c \\ d \end{array} \begin{array}{c} \begin{array}{cccc} a & b & c & d \end{array} \\ \begin{pmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix} \end{array}$$

- But we want to study properties that are independent of the naming, e.g., connectivity.
- Are two given graphs the "same", wrt these properties?

Representing and comparing graphs

We start with simple graphs...



To represent it, we need to name the vertices...

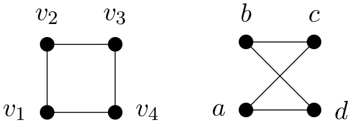
- As an adjacency matrix:

$$\begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \\ v_4 \end{array}\begin{array}{c} \begin{array}{cccc} v_1 & v_2 & v_3 & v_4 \end{array} \\ \begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix} \end{array} \qquad \begin{array}{c} \\ v_1 \\ v_3 \\ v_2 \\ v_4 \end{array}\begin{array}{c} \begin{array}{cccc} v_1 & v_3 & v_2 & v_4 \end{array} \\ \begin{pmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix} \end{array} \qquad \begin{array}{c} \\ a \\ b \\ c \\ d \end{array}\begin{array}{c} \begin{array}{cccc} a & b & c & d \end{array} \\ \begin{pmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix} \end{array}$$

- But we want to study properties that are independent of the naming, e.g., connectivity.
- Are two given graphs the "same", wrt these properties?

Representing and comparing graphs

We start with simple graphs...



To represent it, we need to name the vertices...

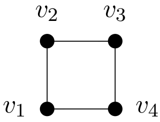
- As an adjacency matrix:

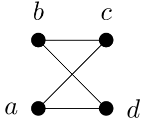
$$\begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \\ v_4 \end{array} \begin{array}{c} \begin{array}{cccc} v_1 & v_2 & v_3 & v_4 \end{array} \\ \begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix} \end{array} \qquad \begin{array}{c} \\ v_1 \\ v_3 \\ v_2 \\ v_4 \end{array} \begin{array}{c} \begin{array}{cccc} v_1 & v_3 & v_2 & v_4 \end{array} \\ \begin{pmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix} \end{array} \qquad \begin{array}{c} \\ a \\ b \\ c \\ d \end{array} \begin{array}{c} \begin{array}{cccc} a & b & c & d \end{array} \\ \begin{pmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix} \end{array}$$

- Reordering of vertices is same as applying a permutation to rows and colums of $A(G)$.
- So, it seems two graphs are "same" if by reordering and renaming the vertices we get the same graph/matrix.

Representing and comparing graphs

We start with simple graphs...





To represent it, we need to name the vertices...

- As an adjacency matrix:

$$
\begin{array}{c@{\,}c}
 & \begin{array}{cccc} v_1 & v_2 & v_3 & v_4 \end{array} \\
\begin{array}{c} v_1 \\ v_2 \\ v_3 \\ v_4 \end{array} & \begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix}
\end{array}
\qquad
\begin{array}{c@{\,}c}
 & \begin{array}{cccc} v_1 & v_3 & v_2 & v_4 \end{array} \\
\begin{array}{c} v_1 \\ v_3 \\ v_2 \\ v_4 \end{array} & \begin{pmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix}
\end{array}
\qquad
\begin{array}{c@{\,}c}
 & \begin{array}{cccc} a & b & c & d \end{array} \\
\begin{array}{c} a \\ b \\ c \\ d \end{array} & \begin{pmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix}
\end{array}
$$

- Reordering of vertices is same as applying a permutation to rows and colums of $A(G)$.
- So, it seems two graphs are "same" if by reordering and renaming the vertices we get the same graph/matrix.
- How do we formalize this?

Isomorphism

Definition

An isomorphism from simple graph G to H is a bijection $f : V(G) \to V(H)$ such that $uv \in E(G)$ iff $f(u)f(v) \in E(H)$.

Isomorphism

Definition

An isomorphism from simple graph G to H is a bijection $f : V(G) \to V(H)$ such that $uv \in E(G)$ iff $f(u)f(v) \in E(H)$.

- Thus, it is a bijection that "preserves" the edge relation.
- Can be checked using adjacency matrix by reordering/renaming.
- What are the properties of this function/relation: $R = \{(G, H) \mid \exists \text{ an isomorphism from } G \text{ to } H\}$.

Isomorphism

Definition

An isomorphism from simple graph G to H is a bijection $f : V(G) \to V(H)$ such that $uv \in E(G)$ iff $f(u)f(v) \in E(H)$.

- Thus, it is a bijection that "preserves" the edge relation.
- Can be checked using adjacency matrix by reordering/renaming.
- What are the properties of this function/relation: $R = \{(G, H) \mid \exists \text{ an isomorphism from } G \text{ to } H\}$.

Proposition

The isomorphism relation is an equivalence relation.

Isomorphism

Definition

An isomorphism from simple graph G to H is a bijection $f : V(G) \to V(H)$ such that $uv \in E(G)$ iff $f(u)f(v) \in E(H)$.

- Thus, it is a bijection that "preserves" the edge relation.
- Can be checked using adjacency matrix by reordering/renaming.
- What are the properties of this function/relation: $R = \{(G, H) \mid \exists \text{ an isomorphism from } G \text{ to } H\}$.

Proposition

The isomorphism relation is an equivalence relation.

- The equivalence classes are called isomorphism classes.

Isomorphism

Definition

An isomorphism from simple graph G to H is a bijection $f : V(G) \to V(H)$ such that $uv \in E(G)$ iff $f(u)f(v) \in E(H)$.

- Thus, it is a bijection that "preserves" the edge relation.
- Can be checked using adjacency matrix by reordering/renaming.
- What are the properties of this function/relation: $R = \{(G, H) \mid \exists \text{ an isomorphism from } G \text{ to } H\}$.

Proposition

The isomorphism relation is an equivalence relation.

- The equivalence classes are called isomorphism classes.
- When we talked about an "unlabeled" graph till now, we actually meant the isomorphism class of that graph!